Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Iterative Learning Control for Periodic Systems using Model Predictive Methods with adaptive sampling rates

Fabian Kennel * **Steven Liu** *

* *Institute of Control Systems, University of Kaiserslautern P.O. Box 3049, 67653 Kaiserslautern, Germany (e-mail: kennel@eit.uni-kl.de, sliu@eit.uni-kl.de).*

Abstract: This paper addresses iterative learning control (ILC) for periodic systems using model predictive and optimization methods to redesign trajectories and reject periodic disturbances. Stability and optimality of these optimization methods is analysed and illustrated on simulations. The additional prospects of the optimization formulation (e.g. including energy costs, system identification) referred to the trajectory planning are accentuated. To reduce the calculation effort of the optimization algorithm a variable and adaptive sampling period is introduced. The advantages compared to classical ILC methods especially in consideration of constraints are presented.

Keywords: Iterative learning control, Model predictive control, Optimization, Periodic control, Trajectory planning, Target tracking, Stability, Disturbance rejection

1. INTRODUCTION

Iterative Learning Control (ILC) (Arimoto et al. [1984] and Moore [1993]) is widely used in industrial repetitive/periodic and iterative processes (*iterative*: robotic automation systems, machine press; *periodic*: motors with eccentric, oscillating steam engines). In general the main idea of this control concept is the disturbance rejection by adapting the reference trajectories. Classical learning approaches transform the trajectories using e.g. P-controllers (Moore [2001], Ratcliffe et al. [2005])

$$u_{j+1} = u_j + Ke_j \tag{1}$$

where u_j is the trajectory of the current iteration, u_{j+1} is the trajectory of the next iteration, e is the tracking error and K is the gain of the ILC. By saving data from the last cycles, calculating new trajectories and applying them to the system, the control error can be reduced iteratively. More information can be found in the survey papers of iterative learning control Wang et al. [2009] and Bristow et al. [2006]. One of the biggest disadvantages of the classical approaches is the absence of a system/disturbance model (model/predictive information) which could significantly improve the control performance.

Many approaches can be found in the literature to solve these problems. ILC strategies using PD/PID controllers (Chen and Moore [2002]/Park et al. [1999], Madady [2008]) include predictive information for a small horizon (one step).

Anticausal filtering algorithms (Verwoerd [2005], van de Wijdeven and Bosgra [2007]) solve the ILC problem using information from the last iterations. The stored data can be referred to the future system behavior and used for the anticausal filter functions. Due to the filter characteristics these methods are unsuitable for changing initial conditions.

Further approaches discuss optimization methods (Pandit and Buchheit [1999], Lee et al. [2000]) which improve the controlled process using system model information. These methods can be divided into two groups: static optimization and dynamic optimization. In general, both approaches are only applicable to cyclic non-periodic systems (non-changing initial condition). The proposed approach in this paper shows how system model data and in addition system limitations (state/input constraints) can be included very efficiently into an ILC design for periodic processes (changing initial conditions) using the beneficial structure of periodic systems (Section 3).

Model predictive control (MPC) ILC approaches which combine the inner control design (process) with the outer ILC strategy can be found in Lee and Lee [2000], Cueli and Bordons [2008] and Wang and Doyle [2009], Chen et al. [2013]. Therefore, for the inner process a MPC has to be designed such that the ILC concept is included. Due to the system dynamic, the prediction horizon of the MPC is limited (calculation effort). A separation of ILC and control design is not given by these concepts which is contradictory to the general idea of ILC: to formulate a general separable approach for controlled cyclic/periodic processes.

For a separable ILC design for periodic processes under constraints new methods have to be developed. The approach presented in this paper concerns the specified issue outlined above using model predictive methods. For this purpose, a minimization problem is introduced and solved such that the stability of the ILC is guaranteed and the learning rate can be adapted continuously without loss of stability. Including MPC methods into the ILC approach leads to planned trajectories considering system constraints. These trajectories are calculated at the beginning of each period. Adapting the cost function of

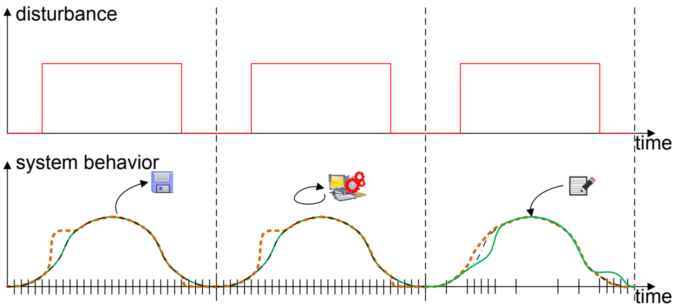

Fig. 1. ILC process

the resulting minimization problem can meet additional optimization objectives (minimization of energy, identification of the system dynamic, adapting inner control parameters). The considered system classes of the approach can be extended to linear time invariant systems with underlying nonlinear systems. To handle large prediction horizons and to reduce the calculation effort, the model predictive strategy uses variable sampling periods. This allows an adaption of the learning process related to the disturbance. In this paper this is called flexible focused learning (FFL).

The paper is organized as follows: Concept and idea of the ILC approach are introduced in Section 2. In Section 3, modelling, optimization, stability and calculation effort of the control concept are presented. In Section 4, an illustrative example is given to demonstrate the prospects of the approach. Finally, Section 5 concludes the paper and accentuates further prospects.

Throughout the paper scalars are indicated by nonbold letters and vectors and matrices by bold letters.

2. CONCEPT

To realize ILC for periodic systems using model predictive methods it is essential to save all state information for one period time. This state information can be used to calculate the optimization step sizes using a step size calculator (reduction of calculation effort). In parallel, the required matrices for the optimization algorithm must be constructed (Section 3). Finally, the optimization problem is solved for the next period and applied to the system. The structure of the ILC process is illustrated in Figure 1 where the system behavior plot shows the reference signal (dash-dotted), the ILC output (solid) and the system response (dashed). For small tracking errors or rather small disturbance variations a large step size is reasonable. If large tracking errors occur, a small step size has to be used around the local tracking error (FFL). The width of the region around the local error is set according to the largest system time constant and the weighting matrices of the optimization function. To keep the ILC concept general, the approach generates trajectories for all state variables (as needed for state control, flatness based control or nonlinear control techniques).

In this paper the method will be illustrated on a state space controlled third order LTI system described by

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}u + \boldsymbol{B}_{\mathrm{d}}d \qquad (2)$$

where $\boldsymbol{x} \in \mathbb{R}^3$ is the state vector, $u \in \mathbb{R}$ is the input and $d \in \mathbb{R}$ is the disturbance. The system is controllable and observable. For simplicity, the approach is described on a single input system. Nevertheless, the theory is also applicable to n-order MIMO systems. The state space controlled process of the example (with ILC) referred to the reference trajectory $\boldsymbol{z}_{\mathrm{nref}}$ can be reformulated to

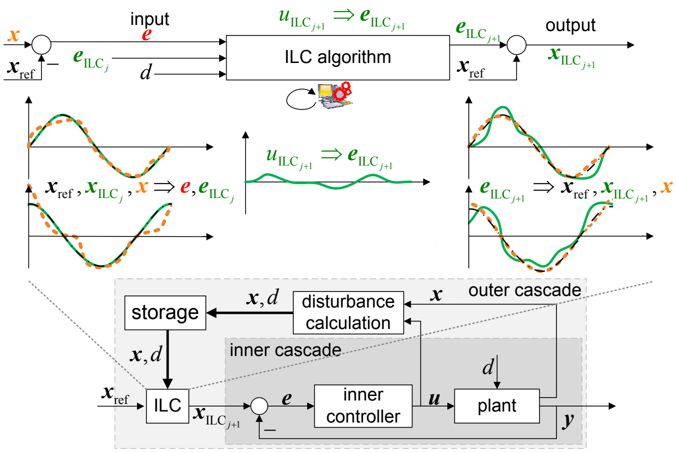

Fig. 2. ILC structure and control structure

$$\dot{\boldsymbol{z}}_{\mathrm{n}} = \boldsymbol{A}_{\mathrm{zn}}\boldsymbol{z}_{\mathrm{n}} + \boldsymbol{B}_{\mathrm{zn}}\underbrace{\left[\boldsymbol{a}_{\mathrm{n}}^T\ 1\right]\begin{bmatrix}\boldsymbol{z}_{\mathrm{nref}}+\boldsymbol{e}_{\mathrm{znILC}}\\ \dot{z}_{\mathrm{n}_n\mathrm{ref}}+\dot{e}_{\mathrm{zn}_n\mathrm{ILC}}\end{bmatrix}}_{u_{\mathrm{nref}}+u_{\mathrm{nILC}}=\left[\boldsymbol{a}_{\mathrm{n}}^T\ 1\right]\boldsymbol{z}_{\mathrm{nILC}}} + \boldsymbol{B}_{\mathrm{zdn}}d_{\mathrm{n}}. \qquad (3)$$

Here, the system is transformed to the controllable canonical form (CCF) with the CCF-coefficients $-\boldsymbol{a}_{\mathrm{n}}^T$ and normalized (with $\boldsymbol{S} = \mathrm{diag}(\boldsymbol{s})$ and the normalization coefficients $\boldsymbol{s}$) to $\boldsymbol{z}_{\mathrm{n}} = \boldsymbol{S}_{\mathrm{z}}\boldsymbol{z}$, $\boldsymbol{A}_{\mathrm{zn}} = \boldsymbol{S}_{\mathrm{z}}\boldsymbol{A}_{\mathrm{z}}\boldsymbol{S}_{\mathrm{z}}^{-1}$, $\boldsymbol{B}_{\mathrm{zn}} = \boldsymbol{S}_{\mathrm{z}}\boldsymbol{B}_{\mathrm{z}}\boldsymbol{S}_{\mathrm{u}}^{-1}$ and $\boldsymbol{B}_{\mathrm{zdn}} = \boldsymbol{S}_{\mathrm{z}}\boldsymbol{B}_{\mathrm{zd}}\ \boldsymbol{S}_{\mathrm{d}}^{-1}$ which is required due to the ILC approach and due to the numerical precision of the algorithm. In addition, a normalization leads to comparable weighting matrices of the optimization problem (Section 4). $u_{\mathrm{nILC}}|\boldsymbol{e}_{\mathrm{znILC}}$ is the additional ILC reference trajectory. $\boldsymbol{z}_{\mathrm{nILC}}$ is the resultant reference trajectory.

In general, the system tracking error dynamic of the periodic process is crucial for the ILC. Hence, the state space dynamic must be referred to. For the presented example, the error dynamic is given by (using Eq. (3))

$$\dot{\boldsymbol{e}}_{\mathrm{n}} = \boldsymbol{A}_{\mathrm{zn}}\boldsymbol{e}_{\mathrm{n}} + \boldsymbol{B}_{\mathrm{zn}}\underbrace{\left[\boldsymbol{a}_{\mathrm{n}}^T\ 1\right]\begin{bmatrix}\boldsymbol{e}_{\mathrm{znILC}}\\ \dot{e}_{\mathrm{zn}_n\mathrm{ILC}}\end{bmatrix}}_{u_{\mathrm{nILC}}} + \boldsymbol{B}_{\mathrm{zdn}}d_{\mathrm{n}}. \qquad (4)$$

The calculated additional reference trajectory of the ILC optimization is given by u_{nILC}. To calculate the corresponding values $\boldsymbol{e}_{\mathrm{znILC}}$, the system dynamic equations

$$\begin{aligned}\dot{\boldsymbol{e}}_{\mathrm{zn}} &= \boldsymbol{A}_{\mathrm{zn}}\boldsymbol{e}_{\mathrm{zn}} + \boldsymbol{B}_{\mathrm{zn}}u_{\mathrm{nILC}} + \boldsymbol{B}_{\mathrm{zdn}}d_{\mathrm{n}}\\ \dot{\boldsymbol{e}}_{\mathrm{znwo}} &= \boldsymbol{A}_{\mathrm{zn}}\boldsymbol{e}_{\mathrm{znwo}} + \boldsymbol{B}_{\mathrm{zdn}}d_{\mathrm{n}}.\end{aligned} \qquad (5)$$

are needed. To get $\boldsymbol{e}_{\mathrm{nILC}}$, the results have to be subtracted.

$$\boldsymbol{e}_{\mathrm{znILC}} = \boldsymbol{e}_{\mathrm{zn}} - \boldsymbol{e}_{\mathrm{znwo}} \qquad (6)$$

For brevity, the proof is omitted. The calculation of $\dot{e}_{\mathrm{zn}_n\mathrm{ILC}}$ results from the derivative of $e_{\mathrm{n}_n\mathrm{ILC}}$. Figure 2 illustrates the ILC structure and the control structure. For the ILC structure (upper figure) the transformation, normalization and state vector computation is done inside the ILC algorithm. j indicates the current period cycle, $\boldsymbol{x}_{\mathrm{ref}}$ is the reference state vector, $\boldsymbol{e}$ is the tracking error, u_{ILC} or rather $\boldsymbol{e}_{\mathrm{ILC}}$ is the additional ILC reference trajectory and $\boldsymbol{x}_{\mathrm{ILC}}$ is the resultant reference trajectory of the system. The control structure (lower figure) shows the relations between plant and controller in the inner cascade and ILC in the outer cascade. A disturbance calculator (model based) reconstructs the periodic system disturbance recursively. All data will be stored and applied to the learning algorithm. The ILC approach itself is separated from the inner control design.

3. ILC-OPTIMIZATION

3.1 Modelling

The central idea of the presented ILC approach is the planning of new trajectories for controlled periodic systems to reject periodic disturbances. In addition, further optimization goals (energy efficiency, considering constraints) can be achieved. For this purpose, a quadratic cost function with constraints is introduced

$$J = \frac{1}{2}\int_0^{T_\mathrm{p}} \boldsymbol{e}_{\mathrm{zn}}^T \boldsymbol{Q}_{\mathrm{zn}} \boldsymbol{e}_{\mathrm{n}} + \tilde{\boldsymbol{u}}_{\mathrm{nILC}}^T \boldsymbol{R}_{\mathrm{n}} \tilde{\boldsymbol{u}}_{\mathrm{nILC}} \mathrm{d}t \qquad (7)$$
$$\boldsymbol{z}_{\mathrm{n}} \in \mathbb{Q}_z, \quad \boldsymbol{e}_{\mathrm{zn}} \in \mathbb{Q}_e, \quad \boldsymbol{u}_{\mathrm{n}} \in \mathbb{Q}_u$$

where T_p is the cycle time of a period and $\tilde{\boldsymbol{u}}_{\mathrm{nILC}}$ is the additional ILC trajectory deviation to a reference ILC trajectory $\boldsymbol{u}_{\mathrm{n\infty ILC}}$. The feasible state variables, errors and learning vectors are set by $\mathbb{Q}_{z|e|u}$. The advantage of an ILC approach using such a weighted ($\boldsymbol{Q}_{\mathrm{zn}}$, $\boldsymbol{R}_{\mathrm{n}}$) cost function, is the expandability, e.g. convex energy terms can be included easily.
Using convex optimization solvers (for constraint problems), the minimization problem can be determined. In this paper, Fast Gradient Methods (FGM) are used (Richter [2012], Kögel and Findeisen [2011]). To calculate the optimization, system and cost function have to be discretized (van Loan [1978] and Franklin et al. [1998]). Using (3) and (7), the cost function can be written as

$$J = \frac{1}{2}\sum_{k=0}^{N-1}\int_{kT_{\mathrm{s}_k}}^{(k+1)T_{\mathrm{s}_k}} \boldsymbol{e}_{\mathrm{zn}}^T \boldsymbol{Q}_{\mathrm{zn}} \boldsymbol{e}_{\mathrm{n}} + \tilde{\boldsymbol{u}}_{\mathrm{nILC}}^T \boldsymbol{R}_{\mathrm{n}} \tilde{\boldsymbol{u}}_{\mathrm{nILC}} \mathrm{d}t = \frac{1}{2}\sum_{k=0}^{N-1} \boldsymbol{v}_k^T \boldsymbol{\mathcal{Q}}_k \boldsymbol{v}_k \qquad (8)$$

where $\boldsymbol{v}_k^T = \left[\boldsymbol{e}_{\mathrm{zn}_k}^T \ \boldsymbol{u}_{\mathrm{n}_{k\mathrm{ILC}}}^T \ \boldsymbol{u}_{\mathrm{n}_{k\infty\mathrm{ILC}}}^T \ \boldsymbol{d}_{\mathrm{n}_k}^T\right]$, T_{s_k} is the sampling period of step k, $\sum_{k=0}^{N-1} T_{\mathrm{s}_k} = T_\mathrm{p}$ and

$$\boldsymbol{\mathcal{Q}} = \int_0^{T_\mathrm{s}} \begin{bmatrix} \boldsymbol{A}_{\mathrm{znd}}^T(\tau) & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{B}_{\mathrm{znd}}^T(\tau) & \boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{B}_{\mathrm{zdnd}}^T(\tau) & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{Q}_{\mathrm{zn}} & & & \\ & \boldsymbol{R}_{\mathrm{n}} & -\boldsymbol{R}_{\mathrm{n}} & \\ & -\boldsymbol{R}_{\mathrm{n}} & \boldsymbol{R}_{\mathrm{n}} & \\ & & & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{A}_{\mathrm{znd}}(\tau) & \boldsymbol{B}_{\mathrm{znd}}(\tau) & \boldsymbol{0} & \boldsymbol{B}_{\mathrm{zdnd}}(\tau) \\ \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \mathrm{d}\tau \qquad (9)$$

where $\boldsymbol{A}_{\mathrm{znd}}(\tau) = e^{\boldsymbol{A}_{\mathrm{zn}}\tau}$, $\boldsymbol{B}_{\mathrm{znd}}(\tau) = \int_0^\tau e^{\boldsymbol{A}_{\mathrm{zn}}\eta}\mathrm{d}\eta \boldsymbol{B}_{\mathrm{zn}}$ and $\boldsymbol{B}_{\mathrm{zdnd}}(\tau) = \int_0^\tau e^{\boldsymbol{A}_{\mathrm{zn}}\eta}\mathrm{d}\eta \boldsymbol{B}_{\mathrm{zdn}}$ and $\boldsymbol{\mathcal{Q}}_{11} = \boldsymbol{Q}_{\mathrm{end}}$, $\boldsymbol{\mathcal{Q}}_{22} = \boldsymbol{R}_{\mathrm{und}}$, $\boldsymbol{\mathcal{Q}}_{33} = \boldsymbol{R}_{\mathrm{u_\infty nd}}$, $\boldsymbol{\mathcal{Q}}_{44} = \boldsymbol{R}_{\mathrm{dnd}}$, $\boldsymbol{\mathcal{Q}}_{12} = \boldsymbol{N}_{\mathrm{eund}}$, $\boldsymbol{\mathcal{Q}}_{13} = \boldsymbol{0}$, $\boldsymbol{\mathcal{Q}}_{14} = \boldsymbol{N}_{\mathrm{ednd}}$, $\boldsymbol{\mathcal{Q}}_{23} = \boldsymbol{N}_{\mathrm{uu_\infty nd}}$, $\boldsymbol{\mathcal{Q}}_{24} = \boldsymbol{N}_{\mathrm{udnd}}$ and $\boldsymbol{\mathcal{Q}}_{34} = \boldsymbol{0}$. Predicting the system dynamics for one period requires a high computational effort which can be reduced using an adaptive step size according to the tracking error (adaptive focused learning), the disturbance dynamic, the weighting matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ and the largest system time constant. In general, the step size should be small around large errors. If T_L is the largest time constant of the system dynamic (related to the disturbance rejection), an adequate band/region is, for instance, T_L (Figure 3). Parallel to the step size calculation, the dynamic system

$$\boldsymbol{E}_{\mathrm{zn}_{j+1}} = \boldsymbol{\Phi}_j \boldsymbol{e}_{\mathrm{zn0}_j} + \boldsymbol{\Gamma}_j \boldsymbol{U}_{\mathrm{nILC}_j} + \boldsymbol{\Xi}_j \boldsymbol{S}_{\mathrm{n}} \qquad (10)$$

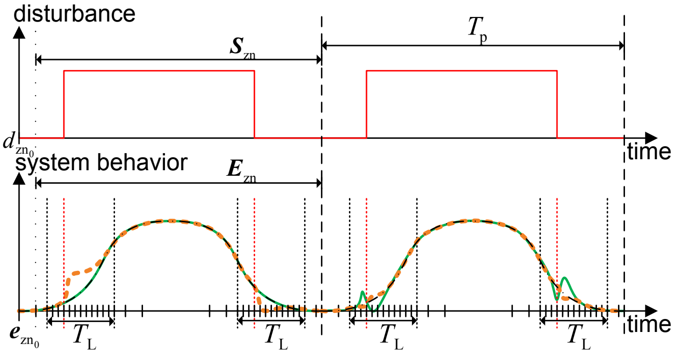

Fig. 3. Step size calculation

can be built up with the system matrices

$$\boldsymbol{\Phi}_j = \begin{bmatrix} \boldsymbol{A}_{\mathrm{znd}_0} \\ \boldsymbol{A}_{\mathrm{znd}_1}\boldsymbol{A}_{\mathrm{znd}_0} \\ \vdots \\ \prod_{k=0}^{N-1}\boldsymbol{A}_{\mathrm{znd}_k} \end{bmatrix}, \boldsymbol{\Gamma}_j = \begin{bmatrix} \boldsymbol{B}_{\mathrm{znd}_0} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{A}_{\mathrm{znd}_1}\boldsymbol{B}_{\mathrm{znd}_0} & \boldsymbol{B}_{\mathrm{znd}_1} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \prod_{k=1}^{N-1}\boldsymbol{A}_{\mathrm{znd}_k}\boldsymbol{B}_{\mathrm{znd}_0} & \prod_{k=2}^{N-1}\boldsymbol{A}_{\mathrm{znd}_k}\boldsymbol{B}_{\mathrm{znd}_1} & \cdots & \boldsymbol{B}_{\mathrm{znd}_{N-1}} \end{bmatrix} \qquad (11)$$

and $\boldsymbol{\Xi}_j$ analog to $\boldsymbol{\Gamma}_j$. $\boldsymbol{E}_{\mathrm{zn}_{j+1}}$, $\boldsymbol{U}_{\mathrm{nILC}_j}$ and $\boldsymbol{S}_{\mathrm{n}}$ contains the state/input and disturbance variables as vectors where the disturbance can be determined from the last steps recursively.
Being periodic, the system is called periodic steady state when $\boldsymbol{e}_{\mathrm{zn}_0} = \boldsymbol{e}_{\mathrm{zn}_N} = \left[\boldsymbol{0}^{n\times(N-1)n} \ \boldsymbol{I}^{n\times n}\right]\boldsymbol{E}_{\mathrm{zn}}$. The limit tracking error trajectory is given by

$$\boldsymbol{E}_{\mathrm{zn}_{\sim j}} = \left(\boldsymbol{I} - \boldsymbol{\Phi}_j^*\right)^{-1}\left(\boldsymbol{\Gamma}_j \boldsymbol{U}_{\mathrm{nILC}_j} + \boldsymbol{\Xi}_j \boldsymbol{S}_{\mathrm{n}}\right) \qquad (12)$$

where $\boldsymbol{\Phi}_j^* = \boldsymbol{\Phi}_j\left[\boldsymbol{0}^{n\times(N-1)n} \ \boldsymbol{I}^{n\times n}\right]$. On the assumption that the disturbance is discrete and periodic and the step size calculation pattern of the periods is constant ($\boldsymbol{\Phi}_{j+1} = \boldsymbol{\Phi}_j = \boldsymbol{\Phi}, \boldsymbol{\Gamma}_{j+1} = \boldsymbol{\Gamma}_j = \boldsymbol{\Gamma}, \boldsymbol{\Xi}_{j+1} = \boldsymbol{\Xi}_j = \boldsymbol{\Xi}$), the limit tracking error trajectory relation of the next to the current cycle can be written as

$$\boldsymbol{E}_{\mathrm{zn}_{\sim j+1}} = \boldsymbol{E}_{\mathrm{zn}_{\sim j}} + \boldsymbol{G}^* \tilde{\boldsymbol{U}}_{\mathrm{nILC}j+1} \qquad (13)$$

where $\boldsymbol{G}^* = (\boldsymbol{I} - \boldsymbol{\Phi}^*)^{-1}\boldsymbol{\Gamma}$ and $\tilde{\boldsymbol{U}}_{\mathrm{nILC}j+1} = \boldsymbol{U}_{\mathrm{nILC}j+1} - \boldsymbol{U}_{\mathrm{nILC}j}$.

3.2 Cost function design and stability

Theorem 1. For the minimization of the discrete cost function [1]

$$J = \min_{\boldsymbol{U}_{\mathrm{nILC}j+1}} \frac{1}{2}\sum_{k=0}^{N-1}\int_{kT_{\mathrm{s}_k}}^{(k+1)T_{\mathrm{s}_k}} \|\boldsymbol{e}_{\mathrm{zn}_{\sim j+1}}\|^2_{\boldsymbol{Q}_{\mathrm{zn}}} + \|\tilde{\boldsymbol{u}}_{\mathrm{nILC}j+1}\|^2_{\boldsymbol{R}_{\mathrm{n}}} \mathrm{d}t = \frac{1}{2}\sum_{k=0}^{N-1} \boldsymbol{v}_k^T \boldsymbol{\mathcal{Q}}_k \boldsymbol{v}_k \qquad (15)$$

[1] formulated as a quadratic function related to $\boldsymbol{U}$:

$$J_{j+1}^* = \min_{\boldsymbol{U}_{\mathrm{nILC}j+1}} \frac{1}{2}\boldsymbol{U}_{\mathrm{nILC}j+1}^T \boldsymbol{G} \boldsymbol{U}_{\mathrm{nILC}j+1} + \boldsymbol{c}^T \boldsymbol{U}_{\mathrm{nILC}j+1} \qquad (14)$$

where $\boldsymbol{G} = \boldsymbol{G}_\mathrm{s}^{*T}\boldsymbol{Q}_{\mathrm{end}}\boldsymbol{G}_\mathrm{s}^* + \boldsymbol{R}_{\mathrm{und}} + 2\boldsymbol{G}_\mathrm{s}^{*T}\boldsymbol{N}_{\mathrm{eund}}$ and $\boldsymbol{c}^T = \boldsymbol{E}_{\mathrm{zns}_{\sim j}}^T\boldsymbol{c}_\mathrm{e} + \boldsymbol{U}_{\mathrm{nILC}_j}^T\boldsymbol{c}_\mathrm{u} + \boldsymbol{S}_\mathrm{n}^T\boldsymbol{c}_\mathrm{d}$ with $\boldsymbol{c}_\mathrm{e} = \boldsymbol{Q}_{\mathrm{end}}\boldsymbol{G}_\mathrm{s}^* + \boldsymbol{N}_{\mathrm{eund}}$, $\boldsymbol{c}_\mathrm{u} = \boldsymbol{N}_{\mathrm{uu_\infty nd}}^T - \boldsymbol{G}_\mathrm{s}^{*T}\boldsymbol{Q}_{\mathrm{end}}\boldsymbol{G}_\mathrm{s}^* - \boldsymbol{G}_\mathrm{s}^{*T}\boldsymbol{N}_{\mathrm{eund}}$, $\boldsymbol{c}_\mathrm{d} = \boldsymbol{N}_{\mathrm{ednd}}^T\boldsymbol{G}_\mathrm{s}^* + \boldsymbol{N}_{\mathrm{udnd}}^T$, $\boldsymbol{G}_\mathrm{s}^* = \boldsymbol{M}\boldsymbol{G}^*$, $\boldsymbol{E}_{\mathrm{zns}_{\sim j}} = \boldsymbol{M}\boldsymbol{E}_{\mathrm{zn}_{\sim j}}$ and $\boldsymbol{M} = \begin{bmatrix} \boldsymbol{0}^{n\times n(N-1)} & \boldsymbol{I}^{n\times n} \\ \boldsymbol{I}^{(n(N-1))\times(n(N-1))} & \boldsymbol{0}^{(n(N-1))\times n} \end{bmatrix}$.

under constraints (18), $\boldsymbol{v}_k^T = \left[\boldsymbol{e}^T_{\mathrm{zn}_{k\sim j+1}}\ \boldsymbol{u}^T_{\mathrm{n}_{k\mathrm{ILC}j+1}}\ \boldsymbol{u}^T_{\mathrm{n}_{k\mathrm{ILC}j}}\ \boldsymbol{d}^T_{\mathrm{n}_k}\right]$, the limit tracking error trajectory $\boldsymbol{e}_{\mathrm{zn}_\sim}$ of the flexible focused iterative learning controlled (FFL-ILC) system is monotonically decreasing from cycle to cycle. The tracking error converges to the periodic steady state error.

Proof 3.1. Setting the cost function of the minimization problem to (15), the limit tracking error trajectory is monotonically decreasing from cycle to cycle caused by (using Eq. (13))

$$\sum_{k=0}^{N-1}\int_{kT_{s_k}}^{(k+1)T_{s_k}} ||\boldsymbol{e}_{\mathrm{zn}_{\sim j+1}}||^2_{\boldsymbol{Q}} + ||\tilde{\boldsymbol{u}}_{\mathrm{nILC}j+1}||^2_{\boldsymbol{R}}\mathrm{d}t \le \sum_{k=0}^{N-1}\int_{kT_{s_k}}^{(k+1)T_{s_k}} ||\boldsymbol{e}_{\mathrm{zn}_{\sim j}}||^2_{\boldsymbol{Q}}\mathrm{d}t. \tag{16}$$

The system tracking error itself converges to the periodic steady state caused to (using Eq. (10) and (12))

$$\tilde{\boldsymbol{E}}_{\mathrm{zn}_{j+1}} = \boldsymbol{\Phi}^* \tilde{\boldsymbol{E}}_{\mathrm{zn}_j} \tag{17}$$

where $\tilde{\boldsymbol{E}}_{\mathrm{zn}_{j|j+1}} = (\boldsymbol{E}_{\mathrm{zn}_{j|j+1}} - \boldsymbol{E}_{\mathrm{zn}_{\sim j}})$ and $\rho(\boldsymbol{\Phi}^*) < 1$ (due to the controlled inner system). To guarantee satisfied constraints of the dynamic system, the gap between the periodic steady state cost function and the dynamic periodic process has to be considered. Therefore, a variable $\boldsymbol{\Delta} > \boldsymbol{0}$ is introduced (constraint conditions) to allow small variations of the current state vector. Hence, satisfied constraints of the ILC are guaranteed. In the presented example the min/max constraints

$$\begin{aligned}
\underline{\boldsymbol{Z}}_{\mathrm{n}} - (\boldsymbol{\Phi}\boldsymbol{\Delta}_{\mathrm{n}})_{\underline{\mathrm{w}}} \le \boldsymbol{Z}_{\mathrm{n}_j} &\le \overline{\boldsymbol{Z}}_{\mathrm{n}} - (\boldsymbol{\Phi}\boldsymbol{\Delta}_{\mathrm{n}})_{\overline{\mathrm{w}}}\\
\underline{\boldsymbol{Y}}_{\mathrm{n}} - (\boldsymbol{\mathfrak{C}}\boldsymbol{\Phi}\boldsymbol{\Delta}_{\mathrm{n}})_{\underline{\mathrm{w}}} \le \boldsymbol{Y}_{\mathrm{n}_j} &\le \overline{\boldsymbol{Y}}_{\mathrm{n}} - (\boldsymbol{\mathfrak{C}}\boldsymbol{\Phi}\boldsymbol{\Delta}_{\mathrm{n}})_{\overline{\mathrm{w}}}\\
\underline{\boldsymbol{U}}_{\mathrm{n}} \le \boldsymbol{U}_{\mathrm{n}_{j+1}} &\le \overline{\boldsymbol{U}}_{\mathrm{n}}\\
-\boldsymbol{\Delta} \le \boldsymbol{z}_{\mathrm{n}_{0_j}} - \boldsymbol{Z}_{\mathrm{n}_{N_{\sim j+1}}} &\le +\boldsymbol{\Delta}\\
-\boldsymbol{\Delta} \le \boldsymbol{z}_{\mathrm{n}_{0_j}} - \boldsymbol{Z}_{\mathrm{n}_{N_j}} &\le +\boldsymbol{\Delta}
\end{aligned} \tag{18}$$

are set, where $\boldsymbol{\mathfrak{C}} = \mathrm{diag}([\boldsymbol{C}_{\mathrm{znd}_1} \cdots \boldsymbol{C}_{\mathrm{znd}_N}])$ is needed to include output constraints $(\underline{\boldsymbol{Y}}, \overline{\boldsymbol{Y}})$, $(\boldsymbol{\Phi}\boldsymbol{\Delta}_{\mathrm{n}})_{\overline{\mathrm{w}}} = \mathrm{abs}(\boldsymbol{\Phi})\boldsymbol{\Delta}$ and $(\boldsymbol{\mathfrak{C}}\boldsymbol{\Phi}\boldsymbol{\Delta}_{\mathrm{n}})_{\overline{\mathrm{w}}} = \mathrm{abs}(\boldsymbol{\mathfrak{C}}\boldsymbol{\Phi})\boldsymbol{\Delta}$ are added for a worst case approximation of the initial state behavior $\boldsymbol{z}_{\mathrm{n}_{0_j}} + \boldsymbol{\epsilon}$ with $|\boldsymbol{\epsilon}| \le \boldsymbol{\Delta}$, $\mathrm{abs}(\cdot)$ describes the absolute values of the insert matrix and $\boldsymbol{Z}_{\mathrm{n}_{N_j}} = \boldsymbol{z}_{\mathrm{n}_{0_{j+1}}}$ is the last state variable of the current period and the initial value of the next period. For all initial values $\boldsymbol{z}_{\mathrm{n}_{0_j}} + \boldsymbol{\epsilon}$ with $|\boldsymbol{\epsilon}| \le \boldsymbol{\Delta}$, the constraints will be satisfied for all periods. Furthermore, $|\boldsymbol{z}_{\mathrm{n}_{0_j}} - \boldsymbol{z}_{\mathrm{n}_{0_{\sim j+1}}}| < \boldsymbol{\Delta}$. Hence, the limit tracking error trajectory of the system satisfies the constraints. Being a controlled LTI system, the process must converge to $\boldsymbol{Z}_{\mathrm{n}_{\sim j+1}}$ (Eq. (17)). Monotonically convergence of the periodic steady state tracking error is guaranteed (Equation (16)). The system converges to the periodic steady state error.□

Due to controllability, observability and the convex optimization, the periodic tracking error converges to zero, if no constraints are violated. The proof is omitted. Stability and constraint relations between the periodic steady state and dynamic system behavior are illustrated in Figure 4 where $\boldsymbol{Z}_{\mathrm{n}_{\sim j}}$ is the periodic steady state of the last cycle, $\boldsymbol{Z}_{\mathrm{n}_j}$ is the current system behavior, $\boldsymbol{Z}_{\mathrm{n}_{\mathrm{ref}}} + \boldsymbol{E}_{\mathrm{znILC}_{j+1}}$ is the ILC output of the current cycle, $\boldsymbol{Z}_{\mathrm{n}_{\sim j+1}}$ is the periodic steady state of the current cycle and $\boldsymbol{Z}_{\mathrm{n}_{\mathrm{ref}}}$ is the reference value.

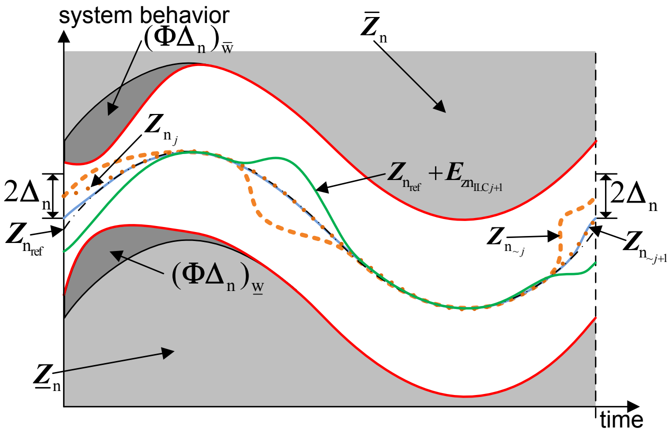

Fig. 4. Periodic steady state/dynamic behavior under constraints

An adaption of $\boldsymbol{R}$ during and between the periods/cycles is practicable without loss of stability. This allows fast learning for the first cycles and slower learning afterwards. In special cases, the learning rate is constrained by $\boldsymbol{\Delta}$. This can be avoided by using a larger $\boldsymbol{\Delta}$ first and a decreasing $\boldsymbol{\Delta}$ from cycle to cycle. Finally, the feasibility of the optimization has to be checked for the first cycle. For all following periods, feasibility is guaranteed automatically.

3.3 *Calculation effort and optimality*

To reduce the calculation effort of the minimization problem, the step size has been adapted (FFL). This has an influence on the quality/costs/optimality of the minimization problem. Therefore, general relations are given (costs with|without adapted step size: $J_{\mathrm{a}}|J_0$):

- large $\boldsymbol{R}$ in relation to $\boldsymbol{Q}$: $J_{\mathrm{a}} \gg J_0$
- small $\boldsymbol{R}$ in relation to $\boldsymbol{Q}$: $J_{\mathrm{a}} \approx J_0$
- high dynamic in d for the hole period: $J_{\mathrm{a}} \gg J_0$
- high dynamic in d only in small regions: $J_{\mathrm{a}} \approx J_0$

Referring to these relations, it is reasonable to increase the step size for a low dynamic in d and a small $\boldsymbol{R}$ and vice versa to decrease the step size for high dynamics in d and a large $\boldsymbol{R}$. Hence, the calculation effort can be reduced without increasing optimization costs (no loss of optimality). For non-changing step size calculations (from cycle to cycle), the required matrices and vectors can be calculated offline. In Kögel and Findeisen [2011] a computation of the warm start Fast Gradient Method (FGM) calculations/calculation time is formulated. Nevertheless, for unfinished optimizations, the last calculated trajectories can be applied to the system.

4. SIMULATION

To illustrate the effect of the presented approach, this section concentrates on the simulation results. The depicted process is a LTI state space controlled ($\boldsymbol{K}$) system in the inner cascade, and the given ILC approach (Section 3) in the outer cascade (Figure 5). All required parameters to specify system, controller, and ILC are listed in Table 1.

For the presented example, a rectangular disturbance d acts on the system, which only injects a high dynamic for a small region (falling/rising edge). For the rest of

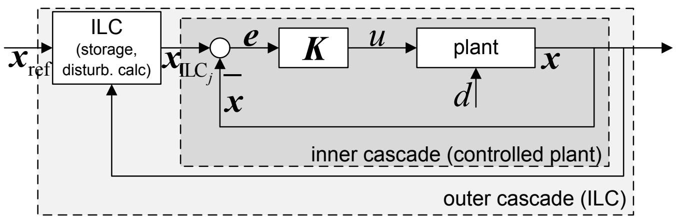

Fig. 5. Cascade structure of the controlled plant

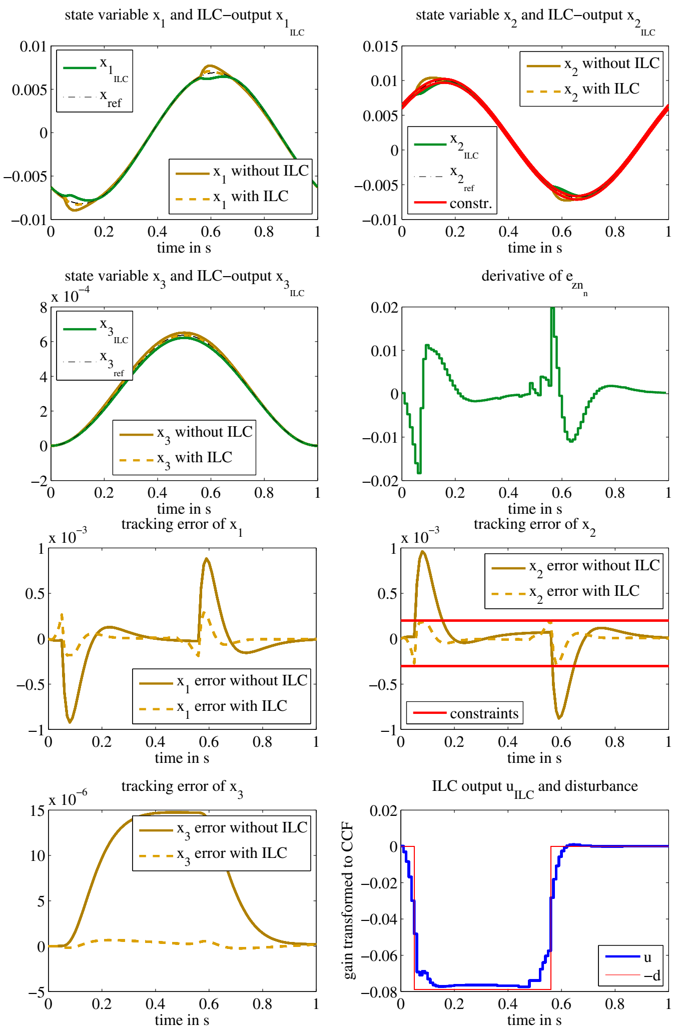

Fig. 6. Simulation results (1)

the period, large sampling times can be applied to the optimization (FFL). The ILC calculates d recursively from the last periods. Sinusoidal reference trajectories are implemented. The simulation results are illustrated in Figure 6 and 7. Figure 6 demonstrates a comparison of a system with and without ILC. It can be seen, that the constraints are satisfied by the optimization. Figure 7 depicts the learning process which starts at the second period. As illustrated, the calculated/estimated disturbance describes the real disturbance sufficiently and the tracking error decreases from period to period. Learning at the second period can be realized due to the fact that the optimization calculation finishes in one sampling step. Due to the sampling period adaption, the length N of the optimization can be kept small. This leads to a small optimization problem. Thus, the minimization can be solved in real-

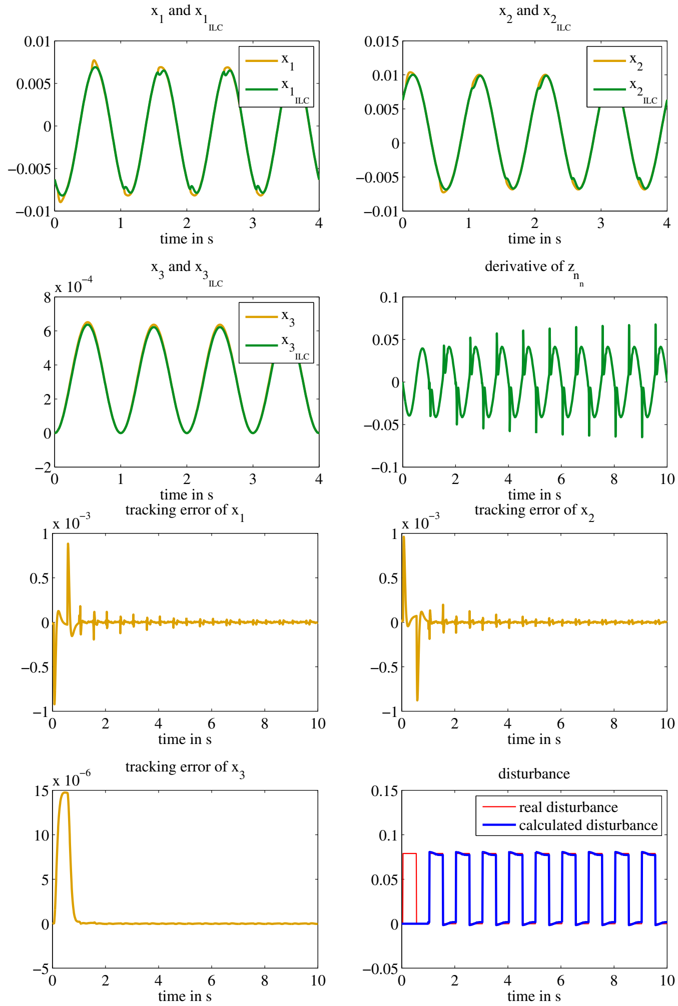

Fig. 7. Simulation results (2)

time. Especially for fast processes and long trajectories (large N) a reduction of the calculation effort is crucial. In Figure 6, the optimization result u_{ILC} is depicted. It is obvious that the sampling period adaption does not influence the result substantially (large step sizes only for a small disturbance dynamic behavior). Comparing the results of the optimization problems (with and without adaptive sampling period), only a cost function deviation in the ppm range occurs while the calculation effort is five times less. $\boldsymbol{\Delta}$ can be set very small without influencing the optimization result (due to the small tracking errors at the beginning of each period).

Flexible and adaptive learning structure due to the dynamic system behavior without loss of stability, consideration of further optimization objectives (energy costs, constraints), identification of the system/disturbance model and adaption of the control parameters are the most additional benefits of the presented approach.

5. CONCLUSIONS

In this paper, iterative learning control for periodic processes using optimization and model predictive methods is presented. The designed cost functions guarantee the stability of the ILC concept. The prospects of including other optimization goals (energy costs, constraints, identification) and dynamic learning (adaptive weighting

Plant		
System matrix	$\boldsymbol{A}$	$\begin{bmatrix} -4 & 1 & 1 \\ 0 & -3 & 1 \\ 1 & 1 & -3 \end{bmatrix}$
Input matrix	$\boldsymbol{B}^T$	$\begin{bmatrix} -1 & -1 & 0 \end{bmatrix}^T$
Disturb. input matrix	$\boldsymbol{B}_{\mathrm{d}}^T$	$\begin{bmatrix} -1 & -1 & 0 \end{bmatrix}^T$
Output matrix	$\boldsymbol{C}$	$\boldsymbol{I}$
Feedthrough matrix	$\boldsymbol{D} = \boldsymbol{D}_{\mathrm{d}}$	$\mathbf{0}$
Sampling periods	$T_{\mathrm{s}}, 4T_{\mathrm{s}}, 8T_{\mathrm{s}}$	0.01, 0.04, 0.08
Periodic time	T_{p}	1
Controller		
State controller (system poles: -20,-22,-24)	$\boldsymbol{K}$	$\begin{bmatrix} 513 & 569 & 3449 \end{bmatrix}$
ILC		
Normalizing $\boldsymbol{z}$	$\boldsymbol{s}_{\mathrm{z}}$	$\begin{bmatrix} 3142 & 1000 & 159 \end{bmatrix}$
Normalizing u	s_{u}	25.3
Normalizing d	s_{d}	0.05
Prediction horizon	N	48
Learning gap	$\boldsymbol{\Delta}$	$\begin{bmatrix} 0.001 & 0.001 & 10^{-7} \end{bmatrix}$
State weighting matrix	$\boldsymbol{Q}_{\mathrm{zn}}$	$\begin{bmatrix} 100000 & 1000 & 10 \end{bmatrix}$
Input weighting matrix	$\boldsymbol{R}_{\mathrm{n}}$	0.5
Min.|Max. tracking error	$\underline{\boldsymbol{e}} \vert \overline{\boldsymbol{e}}$	$\begin{bmatrix} -\infty \\ -3\cdot 10^{-4} \\ -\infty \end{bmatrix} \Bigg\vert \begin{bmatrix} \infty \\ 2\cdot 10^{-4} \\ \infty \end{bmatrix}$
Min.|Max. input	$\underline{u} \vert \overline{u}$	$-\infty \vert 0.3$

Table 1. Simulation parameters

matrices) are accentuated. To make the optimization real-time capable, the calculation effort of the optimization is reduced using variable sampling periods related to the disturbance without loss of optimality.

Future works will focus on the learning algorithm. The learning structure can be expanded to dynamic focused learning where the optimization operates in an activated focus of the period. Furthermore, first order u_{ILC} can be used to decrease the minimization costs especially for large sampling periods. For research, a combination of the presented approach and classical approaches can be fruitful. With a combination, current variations of the system behavior compared to past periods can be included.

REFERENCES

Suguru Arimoto, Sadao Kawamura, and Fumio Miyazaki. Bettering operation of robots by learning. *Journal of Robotic Systems*, 1(2):123–140, 1984.

D.A. Bristow, M. Tharayil, and A. G. Alleyne. A survey of iterative learning control. *IEEE Control Systems Magazine*, 26(3):96–114, June 2006.

Chen Chen, Zhihua Xiong, and Yisheng Zhong. 2-d theory based integrated predictive iterative learning control for batch process. In *2013 10th IEEE International Conference on Control and Automation*, pages 73–77, 2013.

Yang Quan Chen and Kevin L. Moore. An optimal design of pd-type iterative learning control with monotonic convergence. In *Proceedings of the 2002 IEEE Internation Symposium on Intelligent Control Vancover, Canada, Oktober 27-30*, pages 55–60, 2002.

José R. Cueli and Carlos Bordons. Iterative nonlinear model predictive control. stability, robustness and applications. *Control Engineering Practice*, 16(9):1023–1034, 2008.

Gene F. Franklin, J. David Powell, and Michael L. Workman. *Digital Control of Dynamic Systems.* Addison Wesley Longman, 1998.

Markus Kögel and Rolf Findeisen. Fast predictive control of linear, time-invariant systems using an algorithm based on the fast gradient method and augmented lagrange multipliers. In *IEEE International Conference on Control Applications (CCA), Denver, USA, September 28-30*, pages 780–785, 2011.

Jay H. Lee, Kwang S. Lee, and Won C. Kim. Model-based iterative learning control with a quadratic criterion for time-varying linear systems. *Automatica*, 36(5):641–657, 2000.

Kwang S. Lee and Jay H. Lee. Convergence of constrained model-based predictive control for batch processes. *IEEE Transaction on automatic control*, 45(10): 1928–1932, 2000.

Ali Madady. Pid type iterative learning control with optimal gains. *International Journal of Control, Automation, and Systems*, 6(2):194–203, 2008.

Kevin L. Moore. *Iterative Learning Control for Deterministic Systems.* Springer, 1993.

Kevin L. Moore. An observation about monotonic convergence in discrete-time, p-type iterative learning control. In *Proceedings of the 2001 IEEE International Symposium on Intelligent Control, Mexico City, Mexico*, pages 45–49, September 5-7 2001.

M. Pandit and K.-H. Buchheit. Optimizating iterative learning control of cyclic production processes with application to extruders. *IEEE Transaction on Control Systems Technology*, 7(3):382–390, 1999.

Kwang-Hyun Park, Zeungnam Bien, and Dong-Hwan Hwang. A study on the robustness of a pid-type iterative learning controller against initial state error. *International Journal of Systems Science*, 30(1):49–59, 1999.

James D. Ratcliffe, Jari J. Htnen, Paul L. Lewin, Eric Rogers, Thomas J. Harte, and David H. Owens. P-type iterative learning control for systems that contain resonance. *International Journal of adaptive control and signal processing*, 19:769–796, 2005.

Stefan Richter. Computational complexity certification of gradient methods for real-time model predictive control. Dissertation, Department of Informationstechnologie und Elektrotechnik, Eidgenössische Technische Hochschule ETH Zürich, 2012.

Jeroen van de Wijdeven and Okko Bosgra. Noncausal finite-time robust iterative learning control. In *Proceedings of the 46th IEEE Conference on Decision and Control, New Orleans, USA*, December 12-14 2007.

Charles F. van Loan. Computing integrals involving the matrix exponential. *IEEE Transaction on Automatic Control*, 23(3):395–404, 1978.

Markus H. A. Verwoerd. Iterative learning control - a critical review -. Dissertation, Institute of Systems and Control, University of Twente, 2005.

Youqing Wang and Francis J. Doyle. Indirect iterative learning control: Application on artificial pancreatic β-cell. In *Chinese Control and Decision Conference 2009*, pages 1728–1733, 2009.

Youqing Wang, Furong Gao, and Francis J. Doyle. Survey on iterative learning control, repetitive control, and run-to-run control. *Journal of Process Control*, 19(10):1589–1600, 2009.